PATENT
5681-96802/SUN040712

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.: 10/006,089
Filed: December 6, 2001
Inventor:
Gary Cole

Title: System and Method for Managing Information Objects

§ § § § § § § § § § § § § § §

Examiner: YIGDALL, MICHAEL J
Group/Art Unit: 2192
Atty. Dkt. No: 5681-96802

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date indicated below.

Robert C. Kowert
Printed Name

Signature

April 24, 2006
Date

REPLY BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This brief is in reply to the Examiner's Answered dated February 23, 2006. Appellant respectfully requests that this Reply Brief be entered pursuant to 37 C.F.R. § 41.41 and considered by the Board of Patent Appeals and Interferences.

REPLY TO EXAMINER'S ANSWER

Claims 1-4, 6-12 and 14-33 stand finally rejected under 35 U.S.C. § 102(e) as being anticipated by Gwertzman et al. (U.S. Patent 6,189,000) (hereinafter "Gwertzman").

Claims 1-2, 4, 6, 8 -10, 12, 14-18 and 19:

Appellant has argued that Gwertzman does not disclose an identity index that comprises a virtual identity that in turn comprises a plurality of information object identifiers each corresponding to a respective information object, and for each information object, a resource name identifying a resource at which the respective information object is located, wherein the resource name is associated with the respective information object identifier; and wherein the identity index further comprises a resource definition corresponding to each respective named resource, wherein the resource definition further comprises connection information.

As noted previously, Appellant's claimed invention pertains to a particular type of data structure, an identity index, for use in managing user information objects. One embodiment of an identity index is illustrated in Appellant's FIG. 3. The identity index includes a virtual identity (e.g., 312). The virtual identity includes a plurality of information object identifiers (e.g., 350 - JANE_D, janed (352) and JaneD) each corresponding to a respective information object (e.g., 342, 344 and 346). The virtual identity also includes, for each information object, a resource name (e.g., Reso01, Reso02 (353) and Reso03) identifying a resource (210, 212 and 214) at which the respective information object is located, wherein the resource name is associated with the respective information object identifier (e.g., JANE_D, janed (352) or JaneD). The identity index further includes and resource definitions (e.g., 360, 362 and 364), each of which includes connection information (e.g., 368).

Appellant has argued that Gwertzman does not teach a data structure for an identity index as recited in Appellant's claims. In contrast, Gwertzman teaches "a

storage-mechanism interface." In Gwertzman, "instead of having to indicate a path to the storage mechanism and the actual name of the data structure, the application developer needs only to call the data structure a logical name (e.g., 'foo') and the storage-mechanism interface takes care of properly locating and identifying the storage mechanism and the data structure (i.e., providing the actual name of the data structure)" (Gwertzman -- col. 6, lines 59-65). Gwertzman's storage-mechanism interface is a programmatic interface called by application developer code. An application developer using Gwertzman invention would only have to use the logical name for a data structure and Gwertzman's storage-mechanism interface uses the logical name as an index to look up the full path name in a database. Gwertzman has very little relevance to an identity index as recited in Appellant's claim 1.

As noted above, Appellant's claim 1 recites an identity index in which a virtual identity is associated with multiple information objects and resources. Gwertzman does not teach the particular data structure of an identity index that comprises a virtual identity that includes a plurality of information object identifiers. In contrast, each of the entries in Gwertzman's database, which the Examiner equates to virtual identities, contains information regarding only a *single* logical name mapped to a *single* path name. Moreover, as noted above, Gwertzman's system allows a developer to use a logical name rather than a full path or pathname to a data structure. Gwertzman's system does not include multiple information objects associated with a logical name as the Examiner contends. Since, Gwertzman teaches that his system is intended to allow a developer to use a logical name instead of a full pathname to a data structure, it would not make since for Gwertzman's system to include multiple information objects and multiple resource locations for a logical name.

The Examiner cites column 8, lines 28-30 and argues that Gwertzman's database entries may contain a plurality of information object identifiers. However, the cited passage is describing using the DepObject and DepProp fields of the configuration information in TABLE 1. Specifically, Gwertzman teaches that DepObject and DepProp fields may be used to "instantiate a second object using information obtained from a first,

already instantiated object". Here Gwertzman is not describing anything about the entries in his database, which the Examiner equates to an identity index. Instead, Gwertzman is discussing a way to duplicate or create dependencies to an already instantiated *programming* object, especially for use with grouping properties by cross-linking between two storage mechanisms (Gwertzman, column 8, lines 28-42). No mention is made, either at the Examiner's cited passages or elsewhere, regarding a virtual identity that comprises *a plurality* of information object identifiers each corresponding to a respective information object.

In response to Appellant's arguments above, the Examiner argues, in the Response to Arguments section of the Final Action, that Gwertzman's database corresponds to the identity index in Appellant's claim because the database comprises logical names or virtual identities that in turn comprise the actual names of data structures. The Examiner is clearly incorrect. Gwertzman teaches that an entry in his database "includes a field indicating the path name to the storage mechanism associated with the logical name and the actual name of the data structure containing the desired property." Gwertzman's database entry also includes "a field containing a user identity for that storage mechanism or containing a property" (Gwertzman, column 7, lines 1-8). Thus, each entry in Gwertzman's database includes a logical name, a path name (to the storage mechanism), the actual name of the data structure, and a user identity. Each of the entries in Gwertzman's database, which the Examiner equates to virtual identities, contains information regarding only a *single* logical name mapped to a *single* path name. Since, Gwertzman teaches that the purpose of his system is to allow a developer to use a logical name instead of a full pathname to a data structure, it would not make since for Gwertzman's system to include multiple information objects and multiple resource locations for a logical name.

In the Advisory Action, the Examiner refers to the DepObject and DepProp fields of Gwertzman. However, as shown above, Gwertzman teaches that DepObject and DepProp fields may be used to *instantiate different programming objects that each refer to the same stored attribute.* For instance, Gwertzman describes grouping users that all

use the same background color by instantiating multiple objects (e.g. objects representing the user's background color preference) from an already instantiated object (Gwertzman, column 8, lines 28-42). Furthermore, Gwertzman's TABLE 1, cited by the Examiner, only includes a single instance of a DepObject and DepProp fields, not multiple instances, as suggested by the Examiner. Gwertzman's DepObject and DepProp fields have absolutely nothing to do with a virtual identity that includes a plurality of information object identifiers each corresponding to a respective information object, as recited in Appellant's claim 1.

In the Examiner's Answer, the Examiner contends that since a first object is instantiated from Gwertzman's ADSPath and Suffix Fields and that a second object is instantiated using DepObject and DepProp fields, "Gwertzman's database includes identifiers for at least two objects, which is to say a plurality of objects" (Examiner's Answer, page 5, lines 9-17). The Examiner has mischaracterized Gwertzman's teachings. The fields in a given one of Gwertzman's entries are for instantiating a single object. The DepObject and DepProp fields indicate that properties may be inherited from another object when instantiating the current object. Moreover, Appellant's claim 1 recites more than merely a plurality of objects. As noted above, claim 1 recites a plurality of information object identifiers *each corresponding to a respective information object*. As shown above, Gwertzman teaches that DepObject and DepProp fields are part of a configuration file used to instantiate a single programming object (col. 8, lines 3-42). The DepObject and DepProp fields indicate that properties may be inherited from another object when instantiating the current object. The DepObject and DepProp fields clearly cannot be considered the information object identifiers of Appellant's claims that are part of a virtual identity.

Moreover, Appellant's claim 1 recites "for each information object, a resource name identifying a resource at which the respective information object is located, wherein the resource name is associated with the respective information object identifier; and wherein the identity index further comprises a resource definition corresponding to each respective named resource, wherein the resource

definition further comprises connection information". Thus, even if the DepObject and DepProp fields in one of Gwertzman's entries is considered to correspond to the information object identifiers of claim 1, then in order to satisfy the requirements of claim 1, the same entry in Gwertzman would have to include a resource name identifying a resource at which the respective information object identified by the DepObject or DepProp fields is located. However, an entry in Gwertzman clearly include no such information for objects identified by the DepObject and DepProp fields. Further following the Examiner's interpretation, the entry in Gwertzman would also have to include a resource definition corresponding to each respective named resource at which an object identified by the DepObject or DepProp fields is located. An entry in Gwertzman clearly does not include such information for multiple information objects identified in a virtual identity, as recited in Appellant's claim 1.

The Examiner further argues in the Answer that "Gwertzman's ADSPath, Suffix, DepObject and DepProp fields, among others, are indeed included in each entry of the database" in response to Appellant's argument that Gwertzman's DepObject and DepProp fields do not describe anything about the entries in his database (Examiner's Answer, page 5, line 18 – page 6, line 4). However, whether or not each entry in Gwertzman's database includes a DepObject and DepProp fields is irrelevant to the fact that they (the DepObject and DepProp) fields do not each correspond to a respective information object, and thus, cannot correspond to the information object identifiers of Appellant's claim. **Also, as noted above, even if Gwertzman's DepObject and DepProp fields were considered to be information object identifiers each corresponding to a respective information object, in order to meet the limitations of claim 1 the same entry in Gwertzman containing the DepObject and DepProp fields would also have to include a resource definition corresponding to each respective named resource at which an object identified by the DepObject or DepProp fields is located.** The Examiner has not even attempted to explain how this requirement of claim 1 could be met for the DepObject and DepProp fields. The entries in Gwertzman clearly include no such information for multiple information objects identified in a virtual identity, as recited in Appellant's claim 1.

As Gwertzman does not disclose the particular structure of the identity index of Appellant's claimed invention, Gwertzman clearly does not anticipate Appellant's claims.

Therefore, for at lease the reasons above, the rejection of claim 1 is clearly not supported by the cited art and removal thereof is respectfully requested. Similar remarks to those above regarding claim 1 also apply to independent claims 20 and 26, as they include similar limitations to those of claim 1.

Claim 3:

Regarding claim 3, Appellant's have argued that Gwertzman fails to disclose a schema map that maps a resource attribute from the resource to a virtual attribute defined by the schema map. In contrast, as Appellant's have discussed previously, Gwertzman teaches the use of a schema that identifies the properties included within a user object on a server (Gwertzman, column 7, lines 51-65). For example, as Gwertzman describes that if an object includes phone numbers of users, the schema for that object may include an element stating: "phone numbers". Thus, rather than teaching a schema map that maps a resource attribute from a resource to *a virtual attribute defined by the schema map*, Gwertzman teaches a schema that describes what elements or properties are included in a user object. The Examiner cites column 9, lines 28-44 of Gwertzman. However, this portion of Gwertzman does not describe Gwertzman's schema. Instead, the cited passage describes a Get Object function that "is used by an application to obtain an ADS object containing a user property." The cited passage does not even mention any sort of schema or schema map. Nor does it describe mapping a resource attribute to a *virtual attributed defined by a schema map*.

In response to the above arguments, the Examiner, in the Advisory Action, contends that Gwertzman's schema "defines elements that identify the properties included in each information object" and that an element in Gwertzman's schema is a

virtual property or virtual attribute. However, as noted above Gwertzman clearly teaches that the elements of his schema indicate to an application developer that such information is available to access. Gwertzman does not teach a schema map that maps a resource attribute from the resource to a virtual resource attribute defined by the schema map. Instead, as shown above, Gwertzman's schema merely lists the properties of an object that are available for access.

In the Examiner's Answer, the Examiner argues, "contrary to the Appellant's contention, Gwertzman's schema identifies the properties or attributes in each object" (Examiner's Answer, page 7, lines 3-4). The Examiner misunderstood Appellant's argument. As stated above, Appellants are arguing that Gwertzman's schema identifies properties and/or attributes of each object. **The Examiner's statement supports Appellant's argument.** Gwertzman's schema identifies properties and/or attributes of each object and does not map a resource attribute to a *virtual attribute defined by the schema map.*

The Examiner further contends that Gwertzman's "schema 'maps' a property contained in an object to an element such as 'Phone numbers' defined in the schema" (Examiner's Answer, page 7, lines 8-10). Appellant disagrees. Gwertzman clearly teaches that each object includes a schema "which identifies the properties *included within that object*" (Gwertzman, column 7, lines 50-56). Thus, contrary to the Examiner's contention, Gwertzman's schema does not map properties contained in an object to an element defined in the schema. Instead, Gwertzman's schema identifies those actual properties contained in the object so that a developer may then access the property from the object. That is very different from a schema map that maps a resource attribute from the resource to a virtual attributed defined by the schema map. Gwertzman's schema doesn't define virtual attributes as the Examiner contends. Instead, it merely identifies the properties contained in each object.

Thus, for at least the reasons above, the rejection of claim 3 is not supported by the prior art and removal thereof is respectfully requested.

Claim 7:

Appellant has argued that Gwertzman fails to disclose a virtual identity that corresponds to a user. The Examiner cites column 7, lines 14-17 of Gwertzman. However, the cited passage refers to user identification for a storage mechanism containing a property associated with a logical name. Specifically, Gwertzman teaches that a logical name is associated with a data structure containing a desired property and that uniquely identifies that data structure (column 6, lines 52-55). Gwertzman also teaches that the database entry associated with the logical name may include user identification information *for accessing the storage mechanism* that stored the data structure. Thus, the user identification information referred to by the Examiner at column 7, lines 14-17 is describing user identification required to access a storage mechanism associated with a logical name (e.g. virtual identity) and does not describe a logical name or virtual identity that corresponds to a user.

Gwertzman further teaches that such user information is stored in the BindAsName and BindAsPassword entries of TABLE 1. Gwertzman specifically states, [t]he BindAsName and BindAsPassword field[s] are used to tell the storage-mechanism interface the user credentials and passwords that are authentic for a particular storage mechanism" and that "[w]ithout proper authentication, the requesting application cannot access the storage mechanism containing the desired user property" (column 8, lines 42-48). Thus, Gwertzman describes using user credentials to access storage mechanism storing desired properties (properties associated with a logical name) not that a logical name corresponds to a user.

That Gwertzman's database entry includes user identification and authorization information required for accessing the storage mechanism does not imply that the entry *corresponds to a user*. To the contrary the entry clearly corresponds to the data structure containing a desired property. The user identification and authorization information are clearly additional properties of the storage mechanism on which the data structure is stored.

In the Examiner's Answer, the Examiner contends that Appellant has argued that Gwertzman's user identification information does not describe a logical name or virtual identity that corresponds to a user" (Examiner's Answer, page 8, lines 4-6). The Examiner has misunderstood Appellant's argument. Appellants are arguing that Gwertzman's database entry, which the Examiner equates to the virtual identity of Appellant's claims, does not correspond to a user. Instead, Gwertzman's database entry may include user authentication information but that does not imply that the database entry (which the Examiner equates to a virtual identity) corresponds to a user.

The Examiner also states, "it is noted that the plain language of the claim recites merely that 'said virtual identity corresponds to a user'" and that Gwertzman's "user identification information, 'stored in the BindAsName and BindAsPassword entries of TABLE 1" ... is in fact the information with which the logical name or virtual identity corresponds to a user" (Examiner's Answer, page 8, lines 8-12). The Examiner further states that "each entry in Gwertzman's database corresponds to a set of user credentials, or in other words, corresponds to a user whose credentials are specified." However, as noted above, just because Gwertzman's database entry includes, among other things, user access and identification information does not imply that the database entry corresponds to the user. Gwertzman clearly teaches that the database entry corresponds to the "logical name associated with a data structure containing a desired property and uniquely identifies the data structure" and that the logical name "can also be used to identify the storage mechanism rather than a data structure" (Gwertzman, column 6, lines 55-67). Gwertzman does not teach that the database entry corresponds to a user, only that it may include user authorization information for access the data structure associated with the entry. As argued above and previously, the fact that a database entry may include user authentication information does not make the entire entry *correspond* to a user.

Claim 11:

Regarding claim 11, Gwertzman fails to disclose that the schema map maps a resource attribute from the resource to a virtual attribute defined by the schema map. In contrast, Gwertzman teaches the use of a schema that identifies the properties included within a user object on a server (Gwertzman, column 7, lines 51-65). For example, as Gwertzman describes, if an object includes phone numbers of users, the schema for that object may include an element stating: "phone numbers". Thus, rather than teaching a schema map that maps a resource attribute from a resource to a virtual attribute defined by the schema map, Gwertzman teaches a schema that describes what elements or properties are included in a user object. The Examiner cites column 9, lines 28-44 of Gwertzman. However, this portion of Gwertzman is not describing Gwertzman's schema. Instead, the cited passage is describing a Get Object function that "is used by an application to obtain an ADS object containing a user property." The cited passage does not even mention any sort of schema or schema map. Nor does it describe mapping a resource attribute to a virtual attributed *defined by a schema map*. Please refer to the arguments above regarding claim 3 for a more detailed discussion regarding Gwertzman's failure to disclose a schema map that maps a resource attribute from the resource to a virtual resource attribute defined by the schema map.

Claims 20-24:

Regarding claim 20, Gwertzman does not disclose an identity index that comprises a plurality of virtual identities each corresponding to a user, where each virtual identity includes a plurality of information object identifiers each corresponding to a respective information object, and for each information object, a resource name identifying a resource at which the respective information object is located, wherein the resource name is associated with the respective information object identifier; and wherein the identity index further comprises a resource definition corresponding to each respective named resource, wherein the resource definition further comprises connection information, as asserted by the Examiner.

Gwertzman does not teach a data structure for an identity index as recited in Appellant's claims. In contrast, as argued above regarding claim 1, Gwertzman teaches "a storage-mechanism interface." In Gwertzman, "instead of having to indicate a path to the storage mechanism and the actual name of the data structure, the application developer needs only to call the data structure a logical name (e.g., 'foo') and the storage-mechanism interface takes care of properly locating and identifying the storage mechanism and the data structure (i.e., providing the actual name of the data structure)." Gwertzman -- col. 6, lines 59-65. Gwertzman's storage-mechanism interface is a programmatic interface called by application developer code. An application developer using Gwertzman invention would only have to use the logical name for a data structure and Gwertzman's storage-mechanism interface uses the logical name as an index to look up the full path name in a database.

Gwertzman does not teach virtual identities that correspond to users. Instead, Gwertzman teaches that a logical name, which the Examiner equates to a virtual identity, is associated with a data structure containing a desired property (column 6, lines 50-58). Gwertzman does not teach using virtual identities that correspond to users. Gwertzman teaches that a logical name is typically associated with a data structure containing a desired property and that uniquely identifies that data structure (column 6, lines 52-55). Gwertzman also teaches that the database entry associated with the logical name may include user identification information for accessing the storage mechanism that stored the data structure. Thus, the user identification information referred to by the Examiner at column 7, lines 14-17 is describing user identification required to access a storage mechanism associated with a logical name (e.g. virtual identity) and does not describe a logical name or virtual identity that corresponds to a user.

Gwertzman further teaches that such user information is stored in the BindAsName and BindAsPassword entries of TABLE 1. Gwertzman specifically states, [t]he BindAsName and BindAsPassword field[s] are used to tell the storage-mechanism interface the user credentials and passwords that are authentic for a particular storage

mechanism" and that "[w]ithout proper authentication, the requesting application cannot access the storage mechanism containing the desired user property" (column 8, lines 42-48). Thus, Gwertzman describes using user credentials to access storage mechanism storing desired properties (properties associated with a logical name) not that a logical name corresponds to a user.

Gwertzman also does not teach the particular data structure of an identity index that comprises a plurality of virtual identities that in turn comprises a plurality of information object identifiers each corresponding to a respective information object, and for each information object, a resource name identifying a resource at which the respective information object is located, wherein the resource name is associated with the respective information object identifier; and wherein the identity index further comprises a resource definition corresponding to each respective named resource, wherein the resource definition further comprises connection information.

Additionally, each of the entries in Gwertzman's database, which the Examiner equates to virtual identities, contains information regarding only a *single* logical name mapped to a *single* path name. The Examiner cites column 8, lines 28-30 and argues that Gwertzman's database entries may contain a plurality of information object identifiers. However, the cited passage is describing using the DepObject and DepProp fields of the configuration information in TABLE 1. Specifically, Gwertzman teaches that DepObject and DepProp fields may be used to "instantiate a second object using information obtained from a first, already instantiated object". Gwertzman is not describing anything about the entries in his database, which the Examiner equates to an identity index. Instead, Gwertzman is discussing a way to duplicate an already instantiated programming object, especially for use with grouping properties by cross-linking between two storage mechanisms (Gwertzman, column 8, lines 28-42). No mention is made, either at the Examiner's cited passages or elsewhere, regarding a virtual identity that comprises *a plurality* of information object identifiers each corresponding to a respective information object.

Please also refer to the arguments above regarding claims 1 and 7 as they also apply to claim 20.

Anticipation requires the presence in a single prior art reference disclosure of each and every element of the claimed invention, arranged as in the claim. M.P.E.P 2131; *Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co.*, 221 USPQ 481, 485 (Fed. Cir. 1984). The identical invention must be shown in as complete detail as is contained in the claims. *Richardson v. Suzuki Motor Co.*, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). As Gwertzman does not disclose the particular structure of the identity index of Appellant's claimed invention, Gwertzman clearly does not anticipate Appellant's claims.

Therefore, for at lease the reasons above, the rejection of claim 20 is clearly not supported by the cited art and removal thereof is respectfully requested.

Claim 25:

Regarding claim 25, Gwertzman fails to disclose that the schema map maps a resource attribute from the resource to a virtual attribute defined by the schema map. In contrast, Gwertzman teaches the use of a schema that identifies the properties included within a user object on a server (Gwertzman, column 7, lines 51-65). For example, as Gwertzman describes, if an object includes phone numbers of users, the schema for that object may include an element stating: "phone numbers". Thus, rather than teaching a schema map that maps a resource attribute from a resource to a virtual attribute defined by the schema map, Gwertzman teaches a schema that describes what elements or properties are included in a user object. The Examiner cites column 9, lines 28-44 of Gwertzman. However, this portion of Gwertzman is not describing Gwertzman's schema. Instead, the cited passage is describing a Get Object function that "is used by an application to obtain an ADS object containing a user property." The cited passage does not even mention any sort of schema or schema map. Nor does it describe mapping a resource attribute to a virtual attributed *defined by a schema map.* Please refer to the

arguments above regarding claims 3 and 11 for a more detailed discussion regarding Gwertzman's failure to disclose a schema map that maps a resource attribute from the resource to a virtual resource attribute defined by the schema map.

Claims 26-31 and 33:

Appellant has argued that Gwertzman does not disclose storing an identity index including a plurality of information object identifiers corresponding to a set of information objects that define a user, contrary to the Examiner's assertion. The Examiner cites column 7, lines 1-8 and lines 44-50. However, the cited passages only describe using user identification or credentials to access a data structure including a desired property. Gwertzman's logical names correspond to the desired property. The cited passage does not describe a plurality of information object identifiers corresponding to a set of information objects that define a user. Specifically, Gwertzman teaches that a logical name is typically associated with a data structure containing a desired property and that uniquely identifies that data structure (column 6, lines 52-55). Gwertzman also teaches that the database entry associated with the logical name may include user identification information for accessing the storage mechanism that stored the data structure. Thus, the user identification information referred to by the Examiner at column 7, lines 14-17 is describing user identification required to access a storage mechanism associated with a logical name (e.g. virtual identity) and does not describe a logical name or virtual identity that corresponds to a user.

Gwertzman further teaches that such user information is stored in the BindAsName and BindAsPassword entries of TABLE 1. Gwertzman specifically states, [t]he BindAsName and BindAsPassword field[s] are used to tell the storage-mechanism interface the user credentials and passwords that are authentic for a particular storage mechanism" and that "[w]ithout proper authentication, the requesting application cannot access the storage mechanism containing the desired user property" (column 8, lines 42-48).. Thus, Gwertzman describes using user credentials to access storage mechanism storing desired properties (properties associated with a logical name) not that a logical name corresponds to a user.

Gwertzman also fails to disclose associating a resource definition with each information object identifier, wherein each resource definition corresponds to a different one of a *plurality of resources* at which the information object corresponding to the associated information object identifier is located, and wherein each resource definition contains connection information for the corresponding resource. Gwertzman teaches that an entry in his database "includes a field indicating the path name to the storage mechanism associated with the logical name and the actual name of the data structure containing the desired property." Gwertzman's database entry also includes "a field containing a user identity for that storage mechanism or containing a property" (Gwertzman, column 7, lines 1-8). Thus, each entry in Gwertzman's database includes a logical name, a path name (to the storage mechanism), the actual name of the data structure, and a user identity. Thus, each of the entries in Gwertzman's database, which the Examiner equates to virtual identities, contains information regarding only a *single* logical name mapped to a *single* path name.

In the Examiner's Answer, the Examiner argues that claim 26 does not recite that the identity index comprises a virtual identity that in turn comprises a plurality of information object identifies, but instead "merely recites an 'identity index' that itself comprises a 'plurality of information object identifiers'". The Examiner has misunderstood Appellant's argument. As noted above, Appellant's argument is that Gwertzman's database does not include a plurality of information object identifiers corresponding to a set of information objects *that define a user*. Appellant's reference to virtual identities was intended to illustrated that Gwertzman's database entries, which the Examiner previously identified as virtual identities, are not a plurality of information object identifiers corresponding to a set of information object that define a user.

The Examiner cites column 8, lines 3-25 and argues that Gwertzman's database entries may contain a plurality of information object identifiers. However, the cited passage is describing using the DepObject and DepProp fields of the configuration information in TABLE 1. Specifically, Gwertzman teaches that DepObject and DepProp

fields may be used to "instantiate a second object using information obtained from a first, already instantiated object". Gwertzman is not describing anything about the entries in his database, which the Examiner equates to an identity index. Instead, Gwertzman is discussing a way to duplicate an already instantiated programming object, especially for use with grouping properties by cross-linking between two storage mechanisms (Gwertzman, column 8, lines 28-42). No mention is made, either at the Examiner's cited passages or elsewhere, regarding a virtual identity that comprises *a plurality* of resources at which an information object is located.

Therefore, for at lease the reasons above, the rejection of claim 26 is clearly not supported by the cited art and removal thereof is respectfully requested.

Claim 32:

Appellant has argued that Gwertzman fails to disclose that the schema map maps a resource attribute from the resource to a virtual attribute. In contrast, Gwertzman teaches the use of a schema that identifies the properties included within a user object on a server (Gwertzman, column 7, lines 51-65). For example, as Gwertzman describes, if an object includes phone numbers of users, the schema for that object may include an element stating: "phone numbers". Thus, rather than teaching a schema map that maps a resource attribute from a resource to a virtual attribute, Gwertzman teaches a schema that describes what elements or properties are included in a user object. The Examiner cites column 9, lines 28-44 of Gwertzman. However, this portion of Gwertzman is not describing Gwertzman's schema. Instead, the cited passage is describing a Get Object function that "is used by an application to obtain an ADS object containing a user property." The cited passage does not even mention any sort of schema or schema map. Please refer to the arguments above regarding claims 3 and 11 for a more detailed discussion regarding Gwertzman's failure to disclose a schema map that maps a resource attribute from the resource to a virtual resource attribute.

CONCLUSION

For the foregoing reasons submitted in the Appeal Brief and this Reply Brief, it is submitted that the Examiner's rejections of claims 1-4, 6-12 and 14-33 was erroneous. Reversal of the Examiner's decision is respectfully requested.

The Commissioner is authorized to charge any fees that may be due to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5681-96802/RCK. This Reply Brief is submitted with a return receipt postcard.

Respectfully submitted,

Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: April 24, 2006